



(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
18.06.2014 Bulletin 2014/25

(51) Int Cl.:
G06F 3/0482 ^(2013.01) **G06F 3/0486** ^(2013.01)
G05B 19/41 ^(2006.01)

(21) Application number: **13196764.8**

(22) Date of filing: **11.12.2013**

(84) Designated Contracting States:
**AL AT BE BG CH CY CZ DE DK EE ES FI FR GB
GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO
PL PT RO RS SE SI SK SM TR**
Designated Extension States:
BA ME

(72) Inventors:
• **Wang, Albert**
Rochester Hills, MI 48308 (US)
• **Shenoy, Girish B.**
Rochester Hills, MI 48309 (US)
• **Jain, Gaurav**
Rochester Hills, MI 48309 (US)

(30) Priority: **13.12.2012 US 201213714320**

(71) Applicant: **Dassault Systemes Delmia Corp.**
Auburn Hills, MI 48326 (US)

(74) Representative: **Hirsch & Associés**
58, avenue Marceau
75008 Paris (FR)

(54) **Defining object groups in 3D**

(57) A computer-implemented method for defining a group of virtual objects representative of physical objects includes receiving a user input via an input device, wherein the user input relates to at least one virtual object. The method also includes using a processor to determine a

purpose of the user input, modifying an object group with respect to the virtual object based on the purpose of the user input, and storing the relationship between the object group and the object in a memory area.

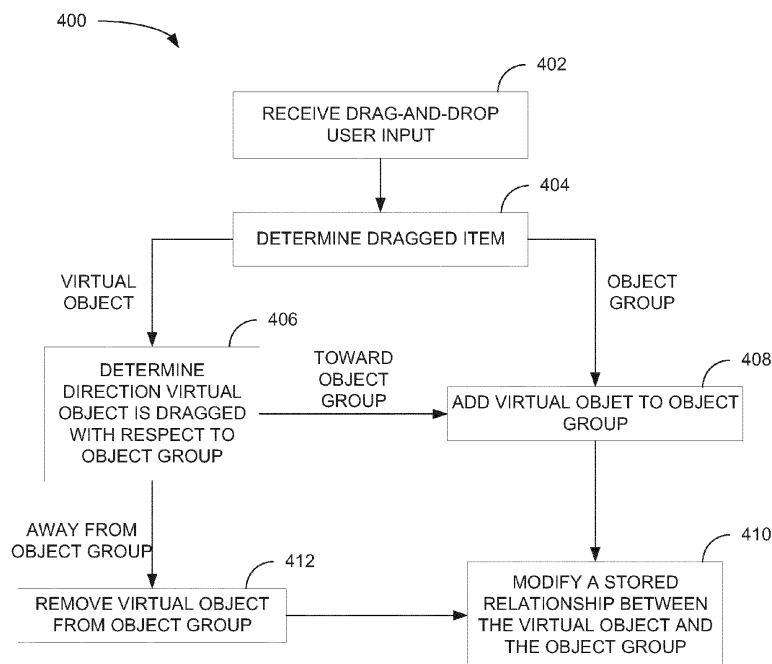


FIG. 4

Description

BACKGROUND OF THE INVENTION

[0001] The embodiments described herein relate generally to managing virtual objects and, more particularly, to creating, modifying, and deleting groups of virtual objects that represent real-world objects.

[0002] Group management is traditionally managed with a table that displays information such as a list of group members, a group name, and elements that are available to add to the group. Moreover, some group management solutions also use images, whether two-dimensional (2D) or three-dimensional (3D) pictures, to illustrate the elements of a group. However, such traditional group management solutions generally include text displays that do not correlate similar elements with advanced technical and/or subjective naming. Moreover, known group management solutions require continuous back-and-forth eye motion from the user between a text display, such as a list of group members, and an associated 2D or 3D representation of the group, its members, and other available elements. This kind of continuous motion can cause fatigue to the user and/or allow the user to more easily make mistakes. In addition, known solutions require continuous back-and-forth motion of an input device, such as moving a mouse back and forth between a list of group members, and an associated 2D or 3D representation of the group, its members, and other available elements. Such motion results in decreased usability and can cause fatigue to the user and/or allow the user to more easily make mistakes.

SUMMARY OF THE INVENTION

[0003] In one aspect, a computer-implemented method is provided for defining a group of virtual objects representative of physical objects. The computer-implemented method includes receiving a user input via an input device, wherein the user input relates to at least one virtual object, and using a processor to determine a purpose of the user input. The method also includes modifying an object group with respect to the virtual object based on the purpose of the user input and storing the relationship between the object group and the virtual object in a memory area.

[0004] The computer-implemented method may further comprise one or more of the following:

[0005] - receiving a user input comprises receiving a first user input representative of a first interaction with the at least one virtual object, and wherein said computer-implemented method further comprises displaying a prompt to the user via a display device to modify the object group with respect to the at least one virtual object;

[0006] - receiving a user input further comprises receiving a second user input representative of a second interaction with the at least one virtual object, and wherein modifying an object group comprises adding the at least

one virtual object to the object group and modifying the stored relationship between the object group and the at least one virtual object in the memory area;

[0007] - modifying the object group further comprises creating the object group into which the at least one virtual object is added;

[0008] - receiving a user input further comprises receiving a second user input representative of a second interaction with the at least one virtual object, and wherein modifying an object group comprises removing the at least one virtual object from the object group and modifying the stored relationship between the object group and the at least one virtual object in the memory area;

[0009] - the user input is a drag-and-drop operation in which the at least one virtual object is dragged toward an indicator of the object group, and wherein modifying an object group comprises adding the at least one virtual object to the object group;

[0010] - the user input is a drag-and-drop operation in which an indicator of the object group is dragged toward the at least one virtual object, and wherein modifying an object group comprises adding the at least one virtual object to the object group.

[0011] In another aspect, a computer device is provided for use in defining a group of virtual objects representative of physical objects. The computer includes an input device, a memory area, and a processor coupled to the input device and the memory area. The processor is configured to receive a user input via the input device, wherein the user input is related to at least one virtual object, and to determine a purpose of the user input. The processor is also configured to modify an object group with respect to the virtual object based on the purpose of the user input and to store the relationship between the object group and the virtual object in the memory area.

[0012] The computer device may further comprise one or more of the following:

- the user input includes a first user input representative of a first interaction with the at least one virtual object, and wherein said processor is further configured to cause a display device to display a prompt to the user to modify the object group with respect to the at least one virtual object;
- the user input further includes a second user input representative of a second interaction with the at least one virtual object, and wherein said processor is configured to add the at least one virtual object to the object group and to modify the stored relationship between the object group and the at least one virtual object in said memory area;
- said processor is further configured to create the object group into which the at least one virtual object is added;
- the user input further includes a second user input

representative of a second interaction with the at least one virtual object, and wherein said processor is configured to remove the at least one virtual object from the object group and to modify the stored relationship between the object group and the at least one virtual object in said memory area;

- the user input is a drag-and-drop operation in which the at least one virtual object is dragged toward an indicator of the object group, and wherein said processor is configured to add the at least one virtual object to the object group;
- the user input is a drag-and-drop operation in which an indicator of the object group is dragged toward the at least one virtual object, and wherein said processor is configured to add the at least one virtual object to the object group.

[0013] In another aspect, a computer system is provided for use in defining a group of virtual objects representative of physical objects. The computer system includes a client device configured to receive a user input related to at least one virtual object. The computer system also includes a server system coupled to the client device. The server system is configured to determine a purpose of the user input, modify an object group with respect to the virtual object based on the purpose of the user input, and store the relationship between the object group and the virtual object in a memory area.

[0014] The computer device may further comprise one or more of the following:

- the user input includes a first user input representative of a first interaction with the at least one virtual object, and wherein said server system is further configured to transmit a signal to said client device representative of a prompt to the user to modify the object group with respect to the at least one virtual object;
- the user input further includes a second user input representative of a second interaction with the at least one virtual object, and wherein said server system is configured to add the at least one virtual object to the object group and to modify the stored relationship between the object group and the at least one virtual object in the memory area;
- said server system is further configured to create the object group into which the at least one virtual object is added;
- the user input further includes a second user input representative of a second interaction with the at least one virtual object, and wherein said server system is configured to remove the at least one virtual object from the object group and to modify the stored

relationship between the object group and the at least one virtual object in the memory area;

- the user input is one of a first drag-and-drop operation in which the at least one virtual object is dragged toward an indicator of the object group and a second drag-and-drop operation in which the indicator of the object group is dragged toward the at least one virtual object, and wherein said server system is configured to add the at least one virtual object to the object group.

[0015] In another aspect, a computer program is provided. The computer program comprises instructions for performing the above method.

BRIEF DESCRIPTION OF THE DRAWINGS

[0016] The foregoing will be apparent from the following more particular description of example embodiments of the invention, as illustrated in the accompanying drawings in which like reference characters refer to the same parts throughout the different views. The drawings are not necessarily to scale, emphasis instead being placed upon illustrating embodiments of the present invention.

Figure 1 is a flowchart that illustrates an exemplary computer-implemented method for defining a group of virtual objects.

Figure 2 is a flowchart that illustrates another exemplary computer-implemented method for defining a group of virtual objects.

Figures 3A-3E are diagrams that illustrate the computer-implemented methods described herein.

Figure 4 is a flowchart that illustrates another exemplary computer-implemented method for defining a group of virtual objects.

Figure 5 is a flowchart that illustrates another exemplary computer-implemented method for defining a group of virtual objects.

Figure 6 is a schematic block diagram of an exemplary computer system for use in defining a group of virtual objects.

Figure 7 is a schematic block diagram of an exemplary computer architecture for use in defining a group of virtual objects.

DETAILED DESCRIPTION OF THE INVENTION

[0017] Exemplary embodiments of computer systems, computer-implemented methods, and computer devices for use in defining and/or managing groups of virtual ob-

jects are described herein. The embodiments described herein facilitate increased productivity for users by enhancing the user's efficiency. For example, the user can create an object group with only a couple of inputs and without scanning a text list of existing groups, elements within the groups, and elements that have not been assigned to any groups. Moreover, the embodiments described herein facilitate decreased eye and/or muscle strain, as well as decreased fatigue, by eliminating the need for the user to continuously move back and forth between a textual representation of group membership and a graphical representation of real-world objects. Furthermore, the embodiments facilitate mobility for users by decreasing the amount of screen necessary to display group memberships and to manage group membership. This enables use of mobile devices such as smartphones and tablet computers.

[0018] As used herein, the term "virtual object" refers generally to a software-generated model of a real-world object. For example, a virtual object may be a three-dimensional model of a machine used in manufacturing. However, the term "virtual object" is not limited to only three-dimensional representations of real-world objects but may also refer to two-dimensional representations. Moreover, the real-world objects are not limited to only manufacturing machines but may instead refer to any suitable resource for use in a predefined process.

[0019] Figure 1 is a flowchart 100 that illustrates an exemplary computer-implemented method for defining a group of virtual objects. In an exemplary embodiment, a computer device receives 102 a user input via an input device, wherein the user input is related to one or more virtual objects. As described above, a virtual object is a computer-generated representation of a physical real-world object. The computer device may be a personal computer, such as a laptop or a desktop computer, or may be a tablet computer or a smartphone. Moreover, the input device may be any suitable device for use in receiving user interactions with displayed objects, such as a mouse, a keyboard, or a touch screen. The user input may be any action taken with respect to the virtual object, such as selecting the virtual object, hovering over the virtual object, and/or a drag-and-drop operation in which the virtual object is manually moved into or out of an object group or elsewhere on a display. Touch inputs may be recognized, received, and/or processed using any suitable means known to those skilled in the art.

[0020] After receiving the user input, the computer device determines 104 a purpose of the user input. Based on the purpose, the computer device modifies 106 an object group with respect to the virtual object. For example, the virtual object may be added to an object group or removed from an object group based on the user input. Moreover, once the object group is modified 106, the computer device stores 108 the relationship between the virtual object and the object group in a memory area. For example, the virtual object and the object group may both be assigned unique identifiers such that, when a relation-

ship between them is created, modified, or destroyed a link is created in a memory area, such as a database, in order to track which virtual objects are associated to which object groups.

[0021] Figure 2 is a flowchart 200 that illustrates another embodiment of a computer-implemented method for defining a group of virtual objects. In an exemplary embodiment, a computer device receives 202 a first user input via an input device, wherein the first user input is related to one or more virtual objects. More specifically, the first user input is a first user interaction with one or more virtual objects. In an exemplary embodiment, the first user input is a use of the input device to hover over the virtual object. For a physical input device such as a mouse, the hover action may be placement of a cursor over or near the virtual object, such as within a predefined area surrounding the virtual object. Alternatively, the hover action may be placement of a cursor over or near an on-screen identifier of the virtual object. For a touch screen device, the hover action may be a touch of the virtual object or an on-screen identifier of the virtual object. For example, the touch may be determined to be a hover action when it is longer than a predefined threshold time. Touch inputs may be recognized, received, and/or processed using any suitable means known to those skilled in the art. In response to the first user input, the computer device displays 204 a prompt on or near the virtual object.

[0022] Figures 3A-3E are diagrams that illustrate the computer-implemented methods shown in Figures 1 and 2, as well as in Figures 4 and 5 as described below. Specifically, Figures 3A-3E show a plurality of virtual objects 300 including a first virtual object 302 through a sixth virtual object 312. As shown in Figures 3A-3E, the embodiments described herein can easily be applied to grouping manufacturing machines that are used to automate at least a portion of a manufacturing process. Such advanced planning of manufacturing processes facilitates increased productivity for a manufacturing plant by enabling advanced simulation of the process from beginning to end, and without creating a physical model or mockup or without using actual machines. It should be noted that the embodiments described herein are not limited to grouping representations of manufacturing machines but, rather, can be used to manage groups of representations of any suitable item.

[0023] Figure 3A shows a starting point in which there are no object groups defined and the virtual objects 300 are not identified on the screen. As shown in Figure 3B, and when the first user input is received 202, the computer device displays a unique identifier 314 for each virtual object 300. For example, the computer device displays a first unique identifier 316 adjacent the first virtual object 302, a second unique identifier 318 adjacent the second virtual object 304, and so on through the sixth unique identifier 326. Figure 3C shows an example of the prompt 328 that is displayed 204 to a user in response to the first user input. Specifically, the prompt 328 shown

in Figure 3C prompts the user to add fifth virtual object 310 to an object group.

[0024] Referring again to Figure 2, the computer device receives 206 a second user input related to the virtual object. More specifically, the second user input is a second user interaction with one or more virtual objects. In an exemplary embodiment, the second user input is a use of the input device to enter a command with respect to the virtual object to modify an object group. For example, the user may select, such as with click or double-click with a mouse or with a touch, tap, or double-tap with a touch screen device, the prompt 328 shown in Figure 3C. As noted above, touch inputs may be recognized, received, and/or processed using any suitable means known to those skilled in the art. The computer device determines 208 the purpose of the second user input and modifies an object group accordingly. When the purpose of the second user input is to add a virtual object to an object group, the computer device determines 210 if an object group exists. If an object group does not exist, the computer device creates 212 a new object group and adds 214 the virtual object to the new object group. If an object group does exist, the computer device adds 214 the virtual object to the object group. Figure 3D shows a newly created object group 330. As described above, the new object group 330 is created by the computer device in order to add the fifth virtual object 310. In an exemplary embodiment, after the virtual object is added 214 to the object group, the computer device modifies 216 a relationship between the virtual object and the object group as stored in a memory area. For example, the virtual object and the object group may both be assigned unique identifiers such that, when a relationship between them is created, modified, or destroyed a link is created in a memory area, such as a database, in order to track which virtual objects are associated to which object groups.

[0025] In an exemplary embodiment, and when the purpose of the second user input is to remove a virtual object from an object group, the computer device removes 218 the virtual object and modifies 216 a relationship between the virtual object and the object group as stored in a memory area. Figure 3C shows the prompt 328 as an addition symbol (+) in order to add the fifth virtual object 310 to an object group. However, if the fifth virtual object 310 were part of an object group, such as object group 330, the prompt 328 is then shown as a subtraction symbol (-) to prompt the user to remove the fifth virtual object 310 from the object group 330.

[0026] In an exemplary embodiment, the above steps can be repeated to add more virtual objects to an object group. For example, in response to another first user interaction the computer device would display 204 a prompt for a virtual object, such as the fourth virtual object 308. In response to another second user interaction the computer device would add 214 the fourth virtual object 308 to an object group, such as object group 330 as shown in Figure 3E.

[0027] Figure 4 is a flowchart 400 of another embodi-

ment of a computer-implemented method for defining a group of virtual objects. In an exemplary embodiment, a computer device receives 402 a drag-and-drop user input. The user input may be received via an input device, such as a mouse or touch pad, or via a touch screen display. Touch inputs may be recognized, received, and/or processed using any suitable means known to those skilled in the art. The computer device then determines 404 the item that was dragged.

[0028] When the item is a virtual object or an indicator or identifier related to a virtual object, the computer device determines 406 the direction the virtual object is dragged with respect to an object group. For example, and referring to Figure 3E, the computer device will determine if an indicator related to the fifth virtual object 310 is dragged out of or away from an indicator related to object group 330, or if an indicator related to the fourth virtual object 308 is dragged towards the indicator related to object group 330. Referring again to Figure 4, if the indicator is dragged toward an object group, the computer device adds 408 the virtual object to the object group. Moreover, the computer device modifies 410 a relationship between the virtual object and the object group as stored in a memory area. For example, the virtual object and the object group may both be assigned unique identifiers such that, when a relationship between them is created, modified, or destroyed a link is created in a memory area, such as a database, in order to track which virtual objects are associated to which object groups. If the indicator is dragged away from an object group, the computer device removes 412 the virtual object from the object group and modifies 410 a relationship between the virtual object and the object group.

[0029] When the item is an object group or an indicator or identifier related to an object group, the computer device adds 408 the virtual object to the object group. Moreover, the computer device modifies 410 a relationship between the virtual object and the object group as stored in a memory area.

[0030] Figure 5 is a flowchart 500 that illustrates another embodiment of a computer-implemented method for defining a group of virtual objects. Referring to Figures 3A-3E and to Figure 5, and in an exemplary embodiment, a user starts 502 a command with an input device or via a touch screen display. Touch inputs may be recognized, received, and/or processed using any suitable means known to those skilled in the art. In response, a computer device displays 504 an identifier with respect to a plurality of virtual objects. Figure 3A shows a plurality of virtual objects 300 prior to the user command and Figure 3B shows the identifiers 314 of each virtual object 300 displayed by the computer device. The computer device then detects 506 an input motion, such as movement of an input device.

[0031] Moreover, the computer device detects 508 when the user places the cursor over or near a virtual object identifier and displays 510 an addition prompt with respect to the virtual object. The placement of the cursor

may be determined by the location as input by an input device or by the location of a tap or contact by the user with a touch screen display. As shown in Figure 3C, when the user positions the cursor over or near fifth identifier 320, for example, a prompt 328 is displayed. This indicates that the virtual object is the currently identified object to manipulate with respect to group membership. The computer device then determines 512 when or if the user drags the virtual object identifier to an existing object group identifier, such as object group 330 (shown in Figure 3D). If the user does not drag the virtual object identifier, then the computer device waits for the next user input. However, if the user drags the virtual object identifier, then the computer device prompts 514 the user to select an object group into which the virtual object is to be added. For example, Figure 3D shows an object group 330 that includes only the fifth virtual object 310. When the user drags the identifier of the fourth virtual object 308, the computer device prompts the user to select a destination object group. The user can then drag the identifier of the fourth virtual object 308 to the identifier of object group 330 in order to add the fourth virtual object 308 to object group 330.

[0032] Furthermore, the computer device may create a new object group into which a virtual object is to be added. In an exemplary embodiment, the computer device detects 516 when the user places the cursor over or near the addition prompt with respect to the virtual object. The placement of the cursor may be determined by the location as input by an input device or by the location of a tap or contact by the user with a touch screen display. The computer device then detects or determines 518 whether the user clicks the addition prompt. If the user does not click the addition prompt, then the computer device waits for the next user input. However, when the user does click the addition prompt, the computer device prompts 520 the user to enter a name for a new object group and adds the virtual object to the new object group and modifies or creates an association in a memory area between the virtual object and the object group. For example, Figure 3C shows prompt 328 related to the fifth virtual object 310. If the computer device detects that the user clicks or selects prompt 328, and when there is no preexisting object group, the computer device prompts the user to enter a new object group name. Figure 3D shows that, in such a case, the user can enter an object group name such as "Group 1" and that the fifth virtual object 310 is added into the new object group.

[0033] The computer device may also remove virtual objects from an object group. In an exemplary embodiment, the computer device detects 522 when the user places the cursor over or near a subtraction prompt with respect to the virtual object. The placement of the cursor may be determined by the location as input by an input device or by the location of a tap or contact by the user with a touch screen display. The computer device then detects or determines 524 whether the user clicks the subtraction prompt. If the user does not click the subtraction

prompt, then the computer device waits for the next user input. However, when the user does click the subtraction prompt, the computer device removes 526 the virtual object from its current object group and modifies or deletes an association in a memory area between the virtual object and the object group.

[0034] Moreover, the computer device may manage existing object groups. In an exemplary embodiment, the computer device detects 528 when the user places the cursor over or near an object group identifier and determines 530 whether a virtual object identifier is selected or grabbed, such as previously grabbed in a drag-and-drop operation in order to add the virtual object to an object group. If the virtual object identifier has been grabbed, the computer device determines 532 whether the identifier has been released. If the identifier has not been released, the computer device waits. When the identifier is released, the computer device adds 534 the virtual object to the object group and modifies or creates an association in a memory area between the virtual object and the object group. Moreover, when the computer device determines 530 that a virtual object identifier has not been selected or grabbed, the computer device displays 536 a removal prompt with respect to the virtual object. The computer device then determines 538 whether an object group identifier is selected such as via double-click or double-tap. If the identifier has not been selected, the computer device waits for the next user input. However, when the identifier is selected, the computer device prompts 540 the user to enter or modify a new identification, such as a name, of the object group. The computer device then saves the new name in the memory area in association with the virtual object(s) that belong to the object group.

[0035] Figure 6 is a schematic block diagram of an exemplary computer system 600 for use in defining a group of virtual objects, and/or for use in performing the processes described above and/or additional processes that may be related to those described above. In an exemplary embodiment, a memory area 602 includes one or more storage devices 604 for use in storing data, such as properties of physical objects, such as manufacturing machines; properties of virtual objects that are used to represent such physical objects; properties of processes in which the physical objects can be used and the associated virtual objects can be used to model; and group membership lists and/or associations that link object groups and virtual objects. The storage devices 604 may be embodied as one or more databases, may be located at a single or at multiple geographical sites, or may be integrated with the server system 606. In some embodiments, the memory area 602 is coupled to a server system 606, which is in turn coupled to client devices such as one or more mobile client devices 608 and/or one or more stationary client devices 610, via a network 612. Mobile client devices 608 may be any mobile device including, but not limited to only including, laptop computers, tablet computers, and/or smartphones. Stationary

client devices 610 may be any device that is generally not mobile, such as desktop computers and the like.

[0036] As can be appreciated, the network 612 can be a public network, such as the Internet, or a private network such as an LAN or WAN network, or any combination thereof and can also include PSTN or ISDN sub-networks. The network 612 can also be wired, such as an Ethernet network, or can be wireless such as a cellular network including EDGE, 3G, and 4G wireless cellular systems. The wireless network can also be WiFi, Bluetooth, or any other wireless form of communication that is known. Thus, the network 612 is merely exemplary and in no way limits the scope of the present advancements.

[0037] The client devices 608 and 610 can be any suitable computer architecture such as the one described below with reference to Figure 7, or any other computing architecture that is known. Moreover, it should be understood that the server system 606 is configured to perform the processes described above and/or any additional processes that may be related to those described above.

[0038] The server system 606 stores the computer-readable instructions to execute the processes described above, and provides these instructions via the network 612 to the client devices 608 and 610. Moreover, the server system 606 can also provide data from the memory area 602 as needed to the client devices 608 and 610 such that the client devices 608 and 610 execute the processes described above. As such, Figure 6 includes implementations of the computer system 600 via cloud computing, distributed computing, and the like.

[0039] Figure 7 is a schematic block diagram of an exemplary computer architecture 700 for use with the server system 606 and/or the client devices 608 and 610 (each shown in Figure 6).

[0040] In an exemplary embodiment, the computer architecture 700 includes one or more processors 702 (CPU) that performs the processes described above and/or any additional processes that may be related to those described above. It should be understood that the term "processor" refers generally to any programmable system including systems and microcontrollers, reduced instruction set circuits (RISC), application-specific integrated circuits (ASIC), programmable logic circuits, and/or any other circuit or processor capable of executing the functions described herein. The above examples are exemplary only and, thus, are not intended to limit in any way the definition and/or meaning of the term "processor."

[0041] The steps of the processes described above and/or any additional processes that may be related to those described above may be stored as computer-executable instructions in, for example, a memory area 704 that is operably and/or communicatively coupled to the processor 702 by a system bus 706. A "memory area," as used herein, refers generally to any means of storing non-transitory program code and instructions executable by one or more processors to aid in defining a group of

virtual objects, and/or for use in performing the processes described above and/or additional processes that may be related to those described above. The memory area 704 may include one, or more than one, forms of memory. For example, the memory area 704 may include random-access memory (RAM) 708, which can include nonvolatile RAM, magnetic RAM, ferroelectric RAM, and/or other forms of RAM. The memory area 704 may also include read-only memory (ROM) 710 and/or flash memory and/or electrically-programmable read-only memory (EEPROM). Any other suitable magnetic, optical, and/or semiconductor memory, such as a hard-disk drive (HDD) 712, by itself or in combination with other forms of memory, may be included in the memory area 704. The HDD 712 may also be coupled to a disk controller 714 for use in transmitting and receiving messages to and from the processor 702. Moreover, the memory area 704 may also be, or may include, a detachable or removable memory 716, such as a suitable cartridge disk, CD-ROM, DVD, or USB memory. The above examples are exemplary only and, thus, are not intended to limit in any way the definition and/or meaning of the term "memory area."

[0042] The computer architecture 700 also includes a display device 718 that is coupled, such as operatively coupled, to a display controller 720. The display controller 720 receives data via the system bus 706 for display by the display device 718. The display device 718 may be, without limitation, a monitor, a television display, a plasma display, a liquid crystal display (LCD), a display based on light-emitting diodes (LED), a display based on organic LEDs (OLED), a display based on polymer LEDs, a display based on surface-conduction electron emitters, a display including a projected and/or reflected image, or any other suitable electronic device or display mechanism. Moreover, the display device 718 may include a touchscreen with an associated touchscreen controller. The above examples are exemplary only and, thus, are not intended to limit in any way the definition and/or meaning of the term "display device."

[0043] In addition, the computer architecture 700 includes a network interface 722 for use in communicating with a network (not shown in Figure 7). Moreover, the computer architecture 700 includes one or more input devices, such as a keyboard 724 and/or a pointing device 726, such as a roller ball, mouse, touchpad, and the like. The input devices are coupled to and controlled by an input/output (I/O) interface 728, which is further coupled to the system bus 706.

[0044] A description of the general features and functionality of the display device 718, keyboard 724, pointing device 726, as well as the display controller 720, disk controller 714, network interface 722, and I/O interface 728 is omitted herein for brevity as these features are known.

[0045] Exemplary embodiments of computer systems, computer-implemented methods, and computer devices for use in defining and/or managing groups of virtual objects are described above in detail. The computer sys-

tems, computer-implemented methods, and computer devices are not limited to the specific embodiments described herein but, rather, operations of the methods and/or components of the system and/or apparatus may be utilized independently and separately from other operations and/or components described herein. Further, the described operations and/or components may also be defined in, or used in combination with, other systems, methods, and/or apparatus, and are not limited to practice with only the systems, methods, and storage media as described herein.

[0046] A computer device, such as those described herein, includes at least one processor or processing unit and a system memory. The computer device typically has at least some form of computer readable media. By way of example and not limitation, computer readable media include computer storage media and communication media. Computer storage media include volatile and nonvolatile, removable and non-removable media implemented in any method or technology for storage of information such as computer readable instructions, data structures, program modules, or other data. Communication media typically embody computer readable instructions, data structures, program modules, or other data in a modulated data signal such as a carrier wave or other transport mechanism and include any information delivery media. Those skilled in the art are familiar with the modulated data signal, which has one or more of its characteristics set or changed in such a manner as to encode information in the signal. Combinations of any of the above are also included within the scope of computer readable media.

[0047] Although the present invention is described in connection with an exemplary computer system environment, embodiments of the invention are operational with numerous other general purpose or special purpose computer system environments or configurations. The computer system environment described herein is not intended to suggest any limitation as to the scope of use or functionality of any aspect of the invention. Moreover, the computer system environment should not be interpreted as having any dependency or requirement relating to any one or combination of components illustrated in the exemplary operating environment. Examples of well known computer systems, environments, and/or configurations that may be suitable for use with aspects of the invention include, but are not limited to, personal computers, server computers, hand-held or laptop devices, multiprocessor systems, microprocessor-based systems, set top boxes, programmable consumer electronics, mobile telephones, network PCs, minicomputers, mainframe computers, distributed computing environments that include any of the above systems or devices, and the like.

[0048] Embodiments of the invention may be described in the general context of computer-executable instructions, such as program components or modules, executed by one or more computers or other devices.

Aspects of the invention may be implemented with any number and organization of components or modules. For example, aspects of the invention are not limited to the specific computer-executable instructions or the specific components or modules illustrated in the figures and described herein. Alternative embodiments of the invention may include different computer-executable instructions or components having more or less functionality than illustrated and described herein.

[0049] The order of execution or performance of the operations in the embodiments of the invention illustrated and described herein is not essential, unless otherwise specified. That is, the operations may be performed in any order, unless otherwise specified, and embodiments of the invention may include additional or fewer operations than those disclosed herein. For example, it is contemplated that executing or performing a particular operation before, contemporaneously with, or after another operation is within the scope of aspects of the invention.

[0050] When introducing elements of aspects of the invention or embodiments thereof, the articles "a," "an," "the," and "said" are intended to mean that there are one or more of the elements. The terms "comprising," "including," and "having" are intended to be inclusive and mean that there may be additional elements other than the listed elements.

[0051] This written description uses examples to disclose the invention, including the best mode, and also to enable any person skilled in the art to practice the invention, including making and using any devices or systems and performing any incorporated methods. The patentable scope of the invention is defined by the claims, and may include other examples that occur to those skilled in the art. Such other examples are intended to be within the scope of the claims if they have structural elements that do not differ from the literal language of the claims, or if they include equivalent structural elements with insubstantial differences from the literal language of the claims.

Claims

1. A computer-implemented method for defining a group of virtual objects representative of physical objects, said computer-implemented method comprising:

receiving (102) a user input via an input device, the user input related to at least one virtual object;

using (104) a processor to determine a purpose of the user input; and

modifying (106) an object group with respect to the at least one virtual object based on the purpose of the user input and storing (108) the relationship between the object group and the at least one virtual object in a memory area.

2. A computer-implemented method in accordance with Claim 1, wherein receiving a user input comprises receiving a first user input representative of a first interaction with the at least one virtual object, and wherein said computer-implemented method further comprises displaying a prompt to the user via a display device to modify the object group with respect to the at least one virtual object.
3. A computer-implemented method in accordance with Claim 2, wherein receiving a user input further comprises receiving a second user input representative of a second interaction with the at least one virtual object, and wherein modifying an object group comprises adding the at least one virtual object to the object group and modifying the stored relationship between the object group and the at least one virtual object in the memory area.
4. A computer-implemented method in accordance with Claim 3, wherein modifying the object group further comprises creating the object group into which the at least one virtual object is added.
5. A computer-implemented method in accordance with Claim 2, wherein receiving a user input further comprises receiving a second user input representative of a second interaction with the at least one virtual object, and wherein modifying an object group comprises removing the at least one virtual object from the object group and modifying the stored relationship between the object group and the at least one virtual object in the memory area.
6. A computer-implemented method in accordance with Claim 1, wherein the user input is a drag-and-drop operation in which the at least one virtual object is dragged toward an indicator of the object group, and wherein modifying an object group comprises adding the at least one virtual object to the object group.
7. A computer-implemented method in accordance with Claim 1, wherein the user input is a drag-and-drop operation in which an indicator of the object group is dragged toward the at least one virtual object, and wherein modifying an object group comprises adding the at least one virtual object to the object group.
8. A computer device for use in defining a group of virtual objects representative of physical objects, said computer comprising:
 - an input device;
 - a memory area; and
 - a processor coupled to said input device and said memory area, said processor configured to
- perform the method according to any one of claims 1 to 7.
9. A computer system for use in defining a group of virtual objects representative of physical objects, said computer system comprising:
 - a client device configured to receive a user input related to at least one virtual object; and
 - a server system coupled to said client device, said server system configured to:
 - determine a purpose of the user input; and
 - modify an object group with respect to the at least one virtual object based on the purpose of the user input and store the relationship between the object group and the at least one virtual object in a memory area.
10. A computer system in accordance with Claim 9, wherein the user input includes a first user input representative of a first interaction with the at least one virtual object, and wherein said server system is further configured to transmit a signal to said client device representative of a prompt to the user to modify the object group with respect to the at least one virtual object.
11. A computer system in accordance with Claim 10, wherein the user input further includes a second user input representative of a second interaction with the at least one virtual object, and wherein said server system is configured to add the at least one virtual object to the object group and to modify the stored relationship between the object group and the at least one virtual object in the memory area.
12. A computer system in accordance with Claim 11, wherein said server system is further configured to create the object group into which the at least one virtual object is added.
13. A computer system in accordance with Claim 10, wherein the user input further includes a second user input representative of a second interaction with the at least one virtual object, and wherein said server system is configured to remove the at least one virtual object from the object group and to modify the stored relationship between the object group and the at least one virtual object in the memory area.
14. A computer system in accordance with Claim 9, wherein the user input is one of a first drag-and-drop operation in which the at least one virtual object is dragged toward an indicator of the object group and a second drag-and-drop operation in which the indicator of the object group is dragged toward the at least one virtual object, and wherein said server sys-

tem is configured to add the at least one virtual object to the object group.

15. A computer program comprising instructions for performing the method of any of claims 1 to 7.

5

10

15

20

25

30

35

40

45

50

55

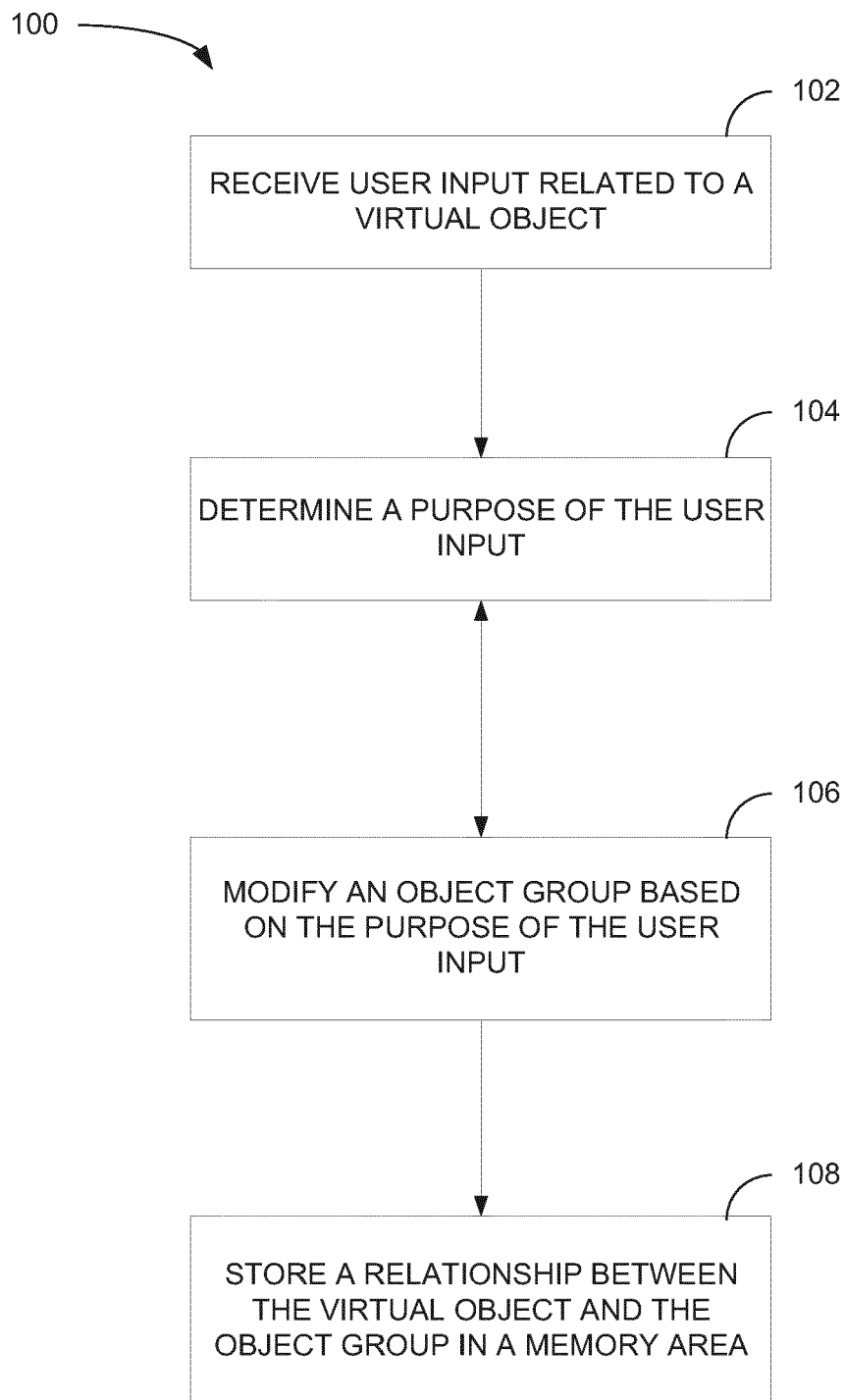
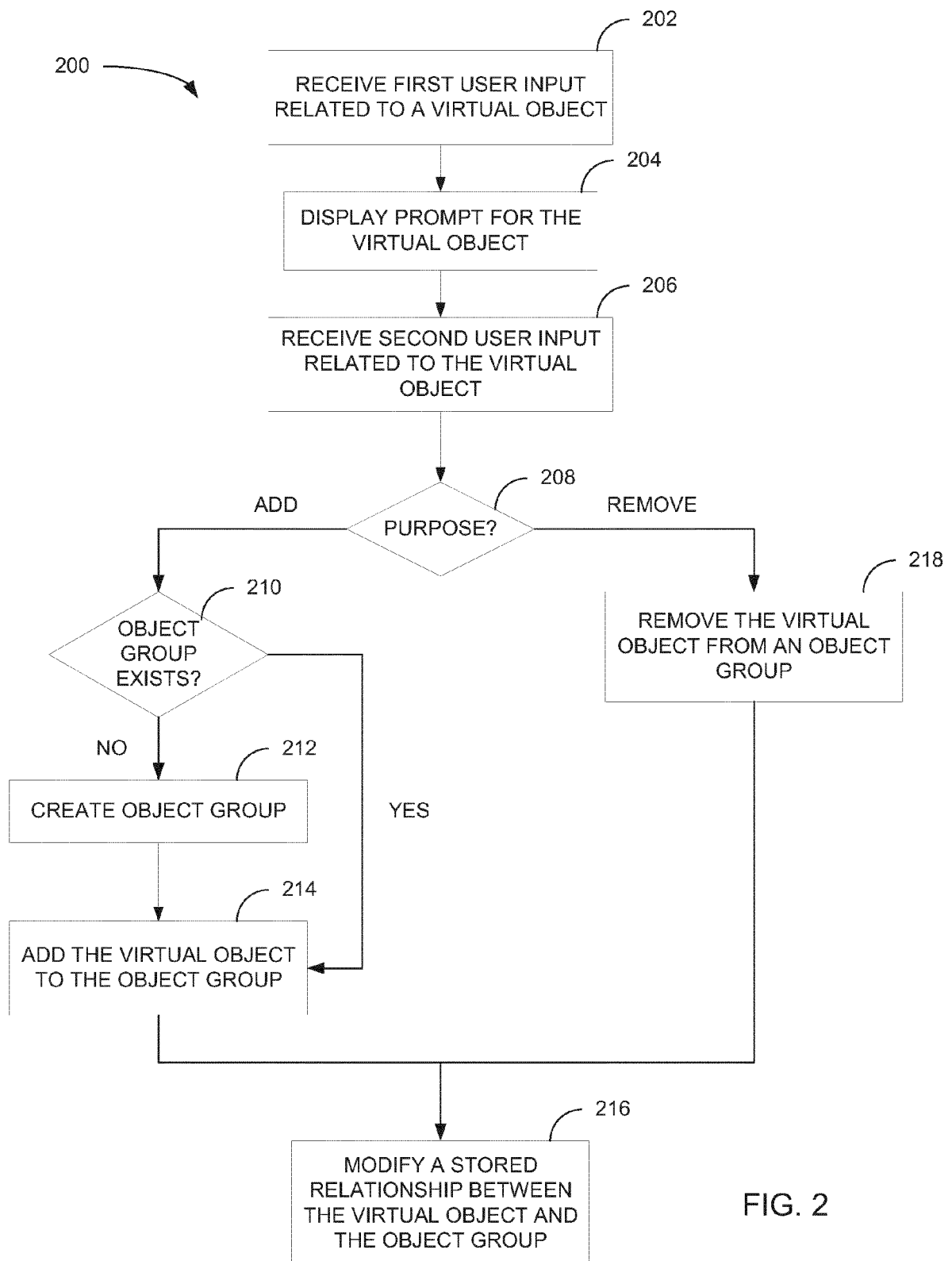


FIG. 1



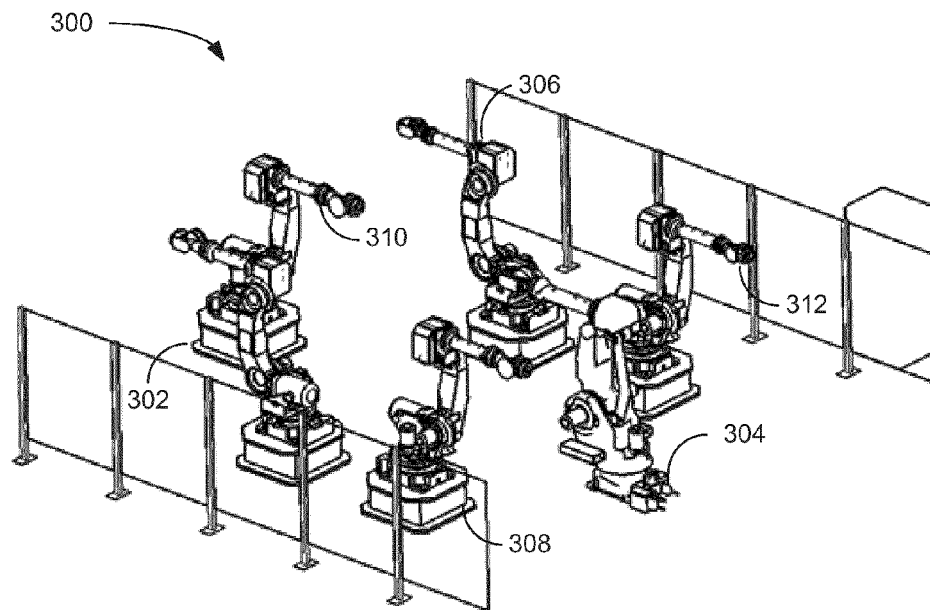


FIG. 3A

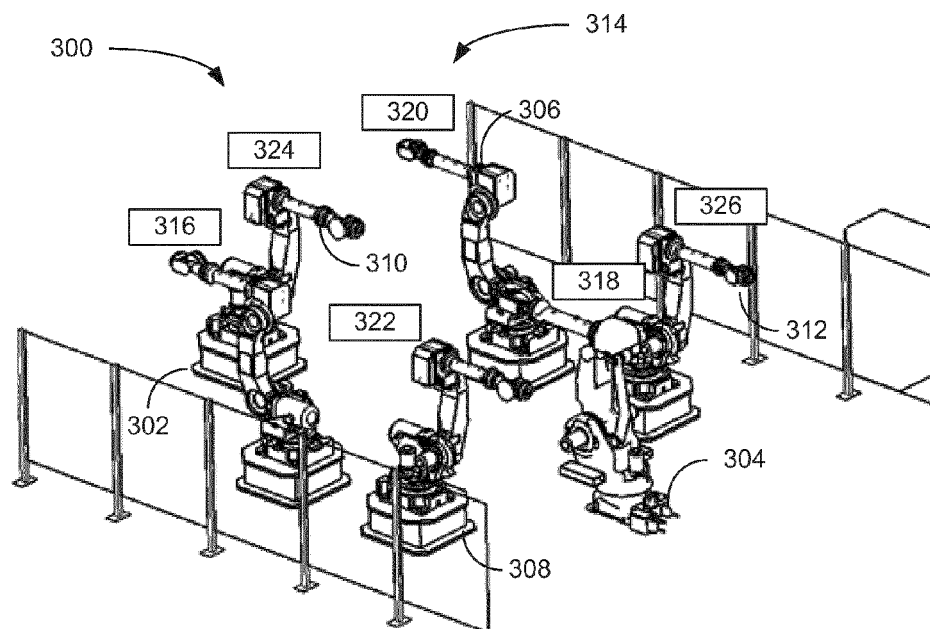


FIG. 3B

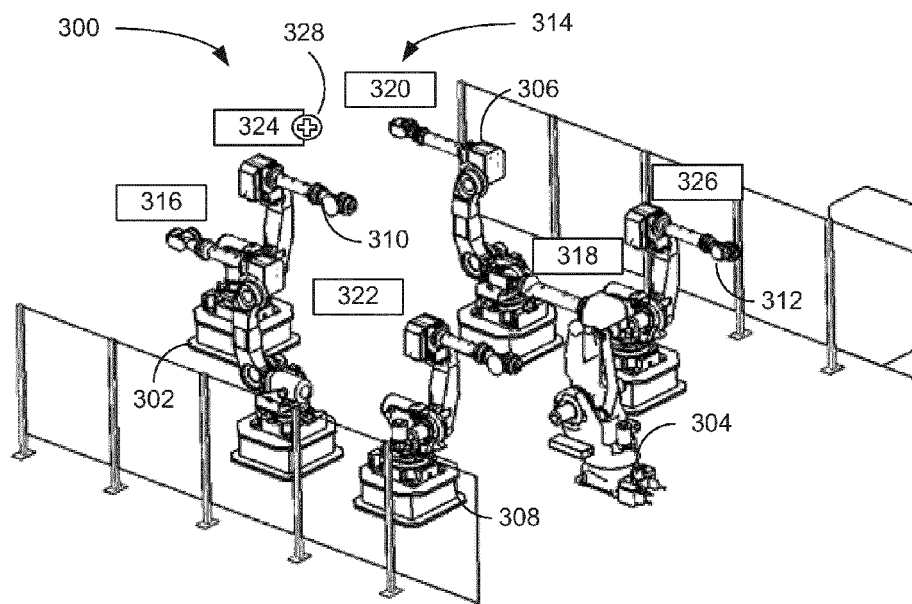


FIG. 3C

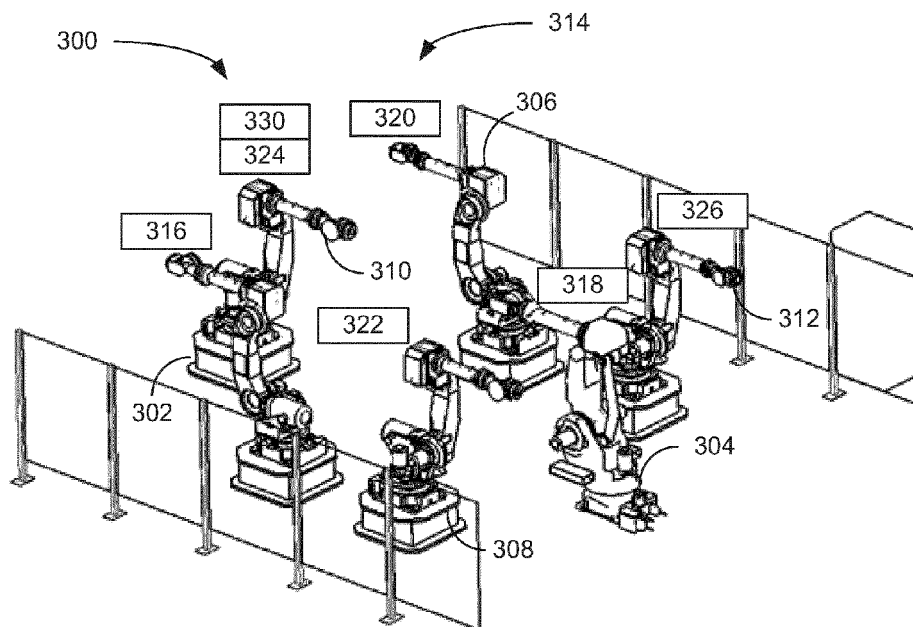


FIG. 3D

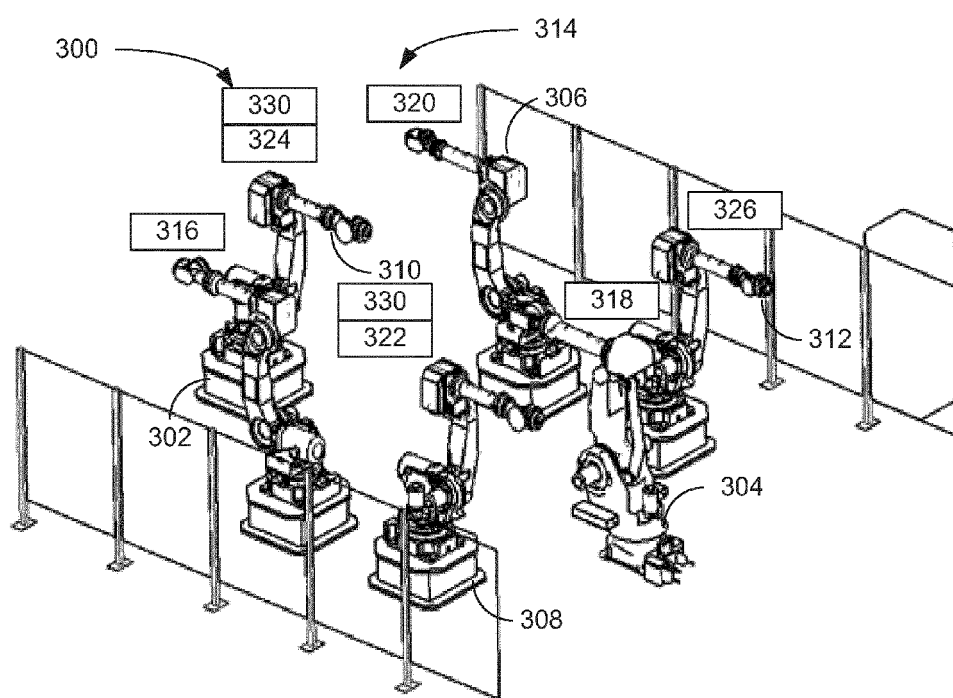


FIG. 3E

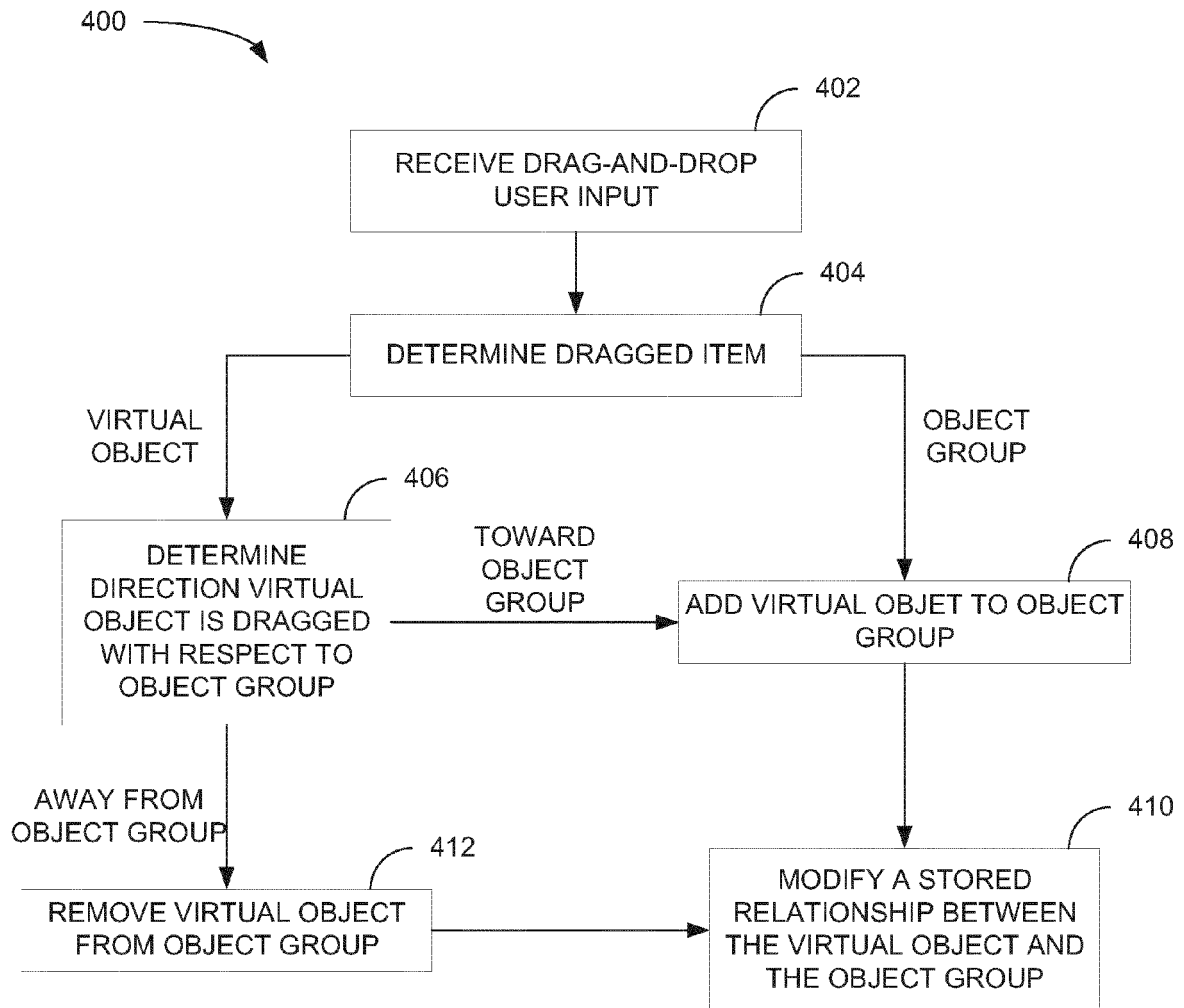
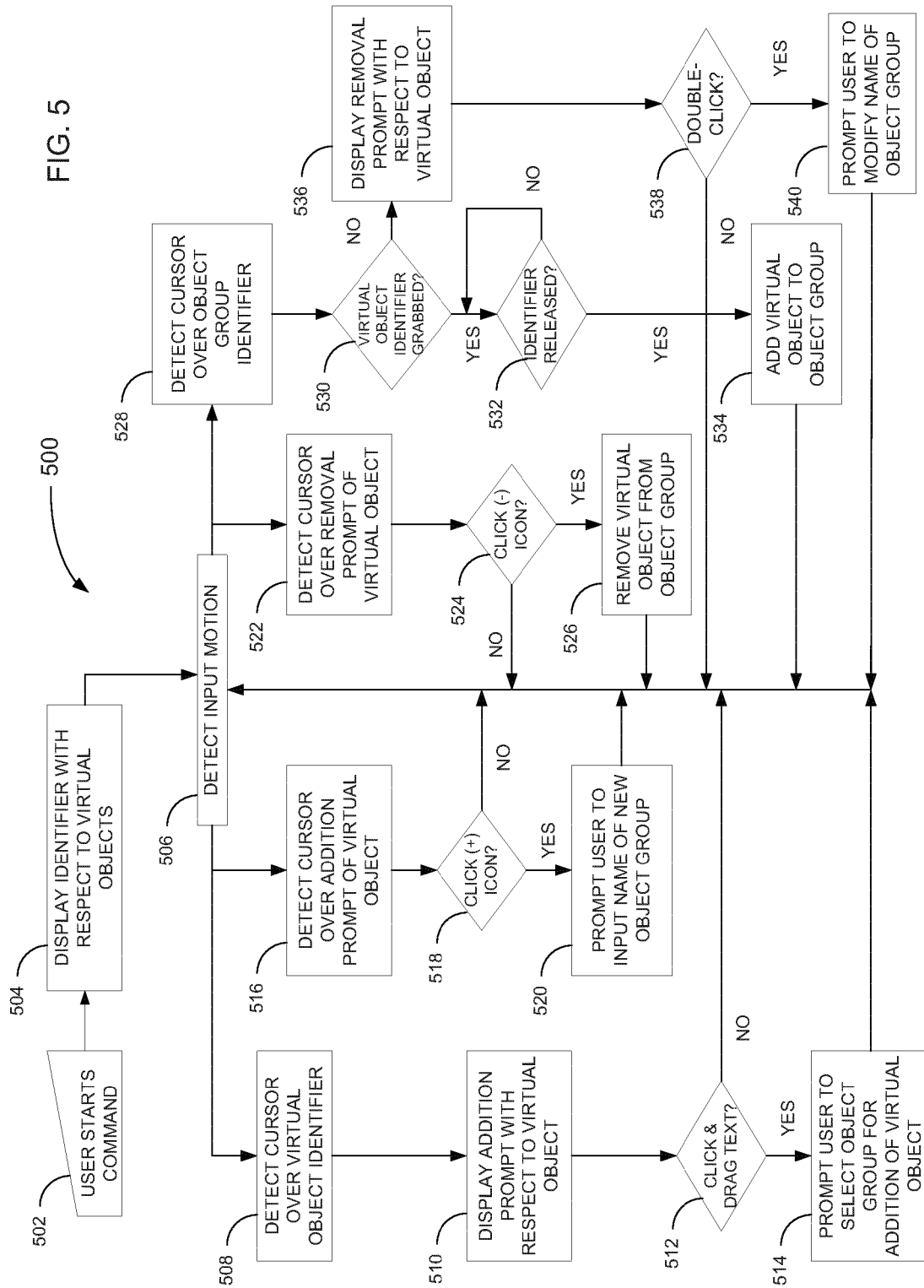


FIG. 4

FIG. 5



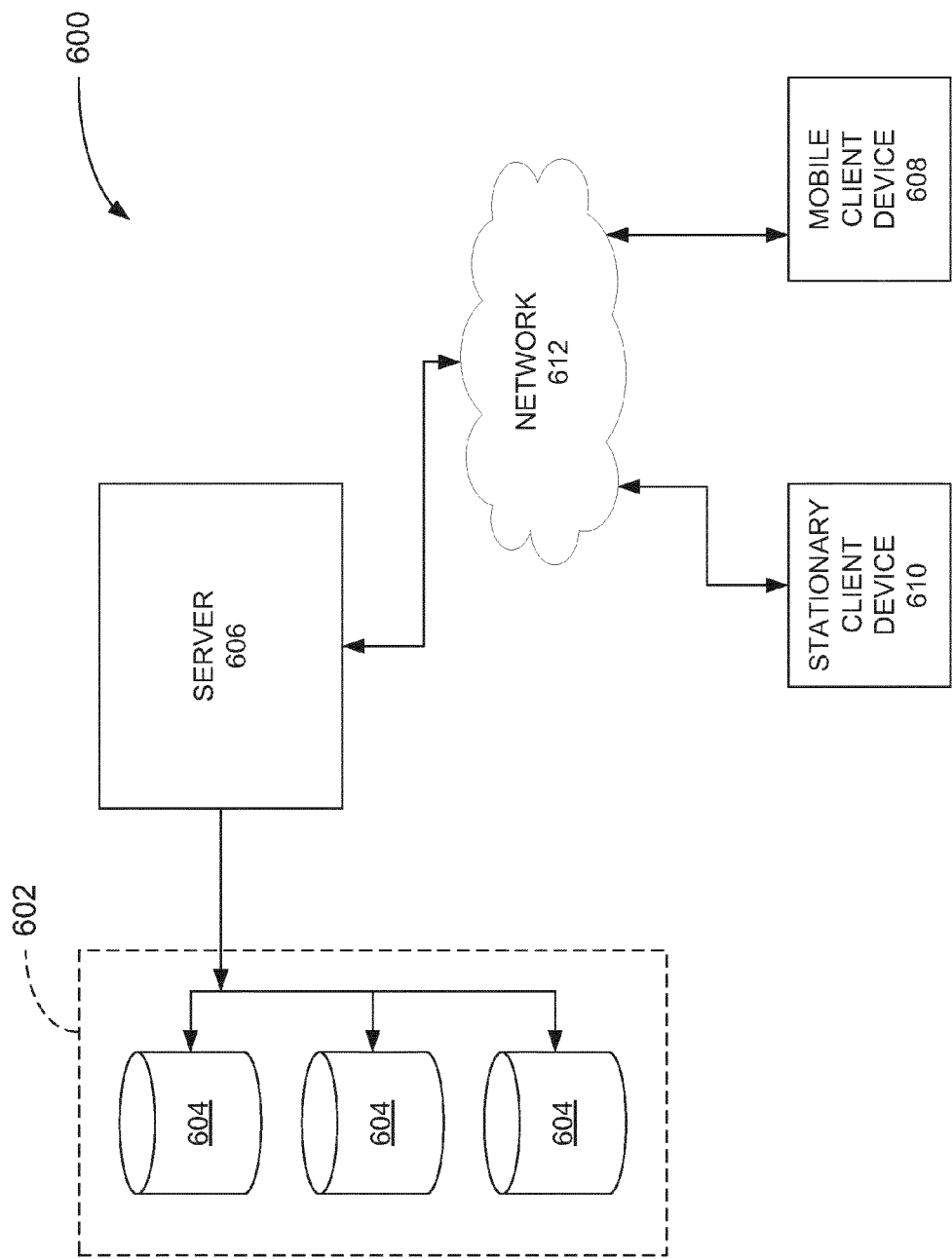


FIG. 6

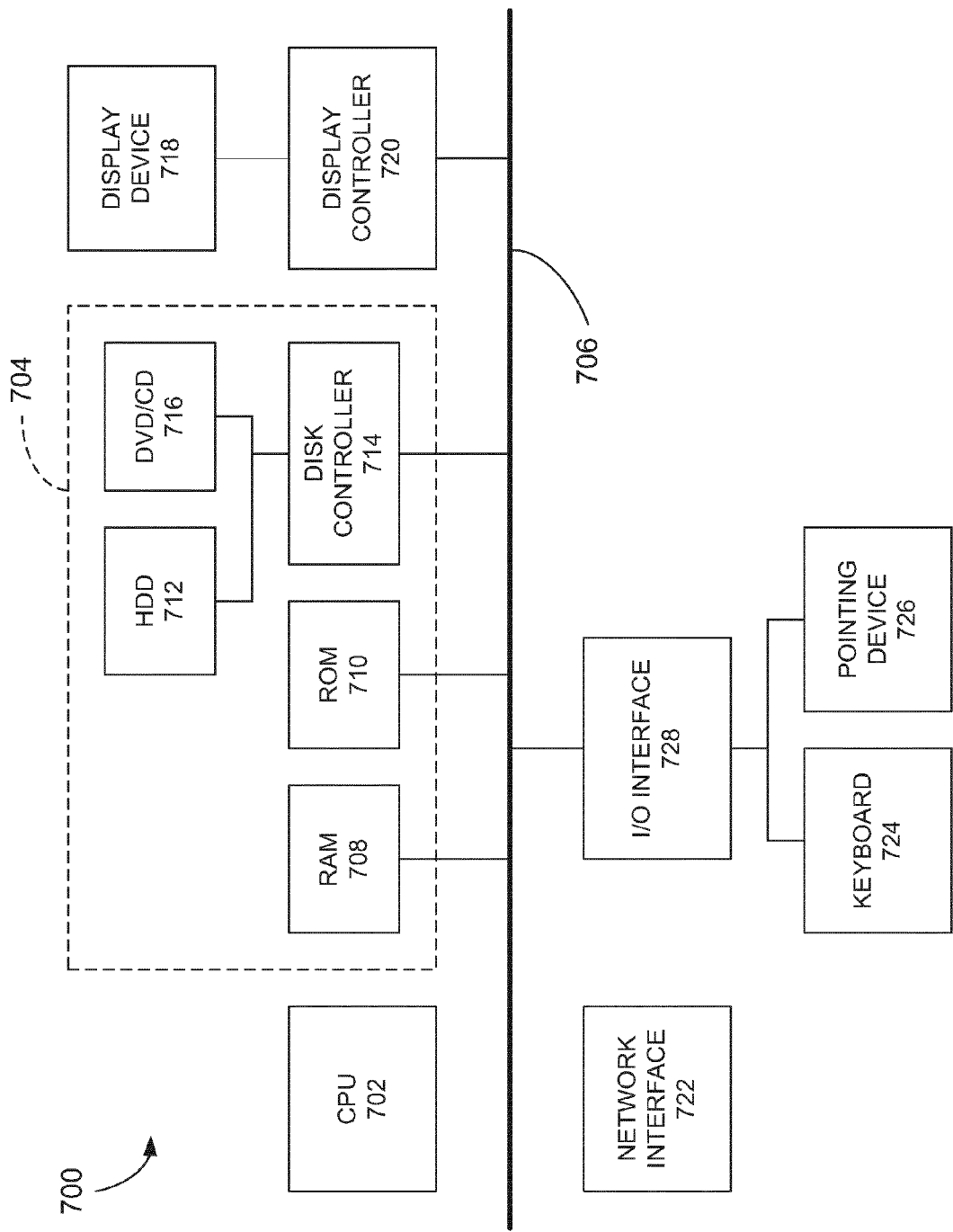


FIG. 7